Interactivity and Multimedia in Case-Based Recommendation

Eoin Hurrell, Alan F. Smeaton and Barry Smyth
CLARITY: Centre for Sensor Web Technologies
School of Computing, Dublin City University, Glasnevin, Dublin 9, Ireland

Abstract

The increasingly prevalent view that recommendation is a conversation between user and system is driving a renewed interest in approaches to system design that involve the user in meaningful ways. In addition to this the proliferation of mobile devices and the near-ubiquity of sensing technologies means that there are now many opportunities to capture real-life experiences, in real-time, providing a new source of raw material for case-based reasoning. In this paper we consider the availability of real-world exercise information, in this cases corresponding to jogging routes, and methods by which we can involve a user in recommending such routes. Outlined here is a novel interaction methodology and compound recommendation system to recommend and adapt routes based on experience not distance/traffic avoidance or similar. We describe the Exercise Builder, a proof-of-concept application that attempts to help visitors to a new city to plan their jogging routes by combining case retrieval, interactive adaptation, and multimedia explanation in a single online service.

Introduction

Recommender systems aim to improve the way by which we access and discover information, products and services and such systems form a key component in e-commerce. To date considerable research has focused on the application of case-based reasoning (CBR) methods in recommender systems (de Mántaras et al. 2005), particularly content-based recommendation.

Generally speaking there are two flavors of recommender system. On the one hand there are *collaborative recommendation systems* (Resnick and Varian 1997), whose recommendations for a target user are based on items that similar users have previously liked. Collaborative recommendation is often referred to as *content-free* because it avoids the need to use item data and instead just relies on the ratings patterns of users across items. In this way collaborative recommendation can suggest movies to a user based solely on the ratings patterns of other users who have watched some of the same movies as the target user, and without the need to understand anything at all about the movies. In contrast, *content-based systems* (Pazzani and Billsus 2007) rely on data about the items that are being recommended; for example, a content-based movie recommender might represent movies according to their genre, actors, director, or producer, recommending new movies to a target user that are similar to other movies that s/he has liked in the past. Ideas from case-based reasoning are commonly used in content-based recommender systems.

To a large extent, the role of CBR in recommender systems has been somewhat limited, focusing on the retrieval of items, rather than the *re-use of experience* which is the core of CBR methods; certainly we seldom see examples of *adaptation* in recommender scenarios, for example. Further to this, new interests in interactive recommendation, and even generally framing recommendation as a conversation (such as presented by (Tunkelang 2011)) has shown considerable fruit, although not without some degree of domain knowledge (Knijnenburg, Reijmer, and Willemsen 2011). All this makes it interesting to consider recommender system applications that provide an opportunity to manipulate experience rather than to simply retrieve items or artifacts, leveraging "known good" cases with a need to explore and adapt, without detailed attribute knowledge. Indeed, it is timely that we begin to consider such notions more seriously because the proliferation of mobile devices and ubiquitous sensing technologies now provide a platform for capturing experience in the real-world (Smyth 2009).

We have developed a framework and a demonstrator system which can solicit useful information from a user and modify the initial recommendation iteratively. Interactive recommendation is not a new idea and has been used in the past in many ways, including as a game to help users discover their true interests in a system, then sharing the information with others (Alon et al. 2009). It has also been proposed as a way of preventing information overload in areas such as e-commerce (Shimazu 2001).

In this paper we consider one approach to the problem of recommending from a case-base that allows for direct user exploration and modification, made manifest as an application in the personal exercise domain. Simply put, the task is to recommend new jogging routes to visitors to an unfamiliar city, given minimal information about their requirements, such as a start location and a preferred distance. The *case* *base* of prior experience is made up of the routes created by and followed by local users, tracked by GPS, and automatically uploaded to a popular run-recording website. In this paper we describe a conversational recommender system in the style of (McGinty and Smyth 2002), whereby the recommender system focuses on the route selection/retrieval and provides an opportunity for the user to interactively adapt the suggested route. We focus on how routes can be retrieved based on a location-popularity metric calculated across the case base, and we further describe a technique for extending and combining routes that individually do not meet the user's needs. This provides a recommendation to the user that represents the judgement of others experienced with running in the locale would factoring in elements such as traffic and terrain.

Thus by allowing the user to interactively adjust the recommended route, we are effectively re-introducing adaptation into case-based recommendation and given that the use-case of this system is for a visitor to a new city, we have also developed an explanation component as part of this.

Recommending Routes in a New City

The Exercise Builder system we developed to test the concept of re-using experience through CBR is in itself novel, as most applications of recommender systems for health-related fields have been designed to aid doctors, such as seen in (Miyo et al. 2007). To our knowledge, a recommender concerned with fitness and the reuse of knowledge provided by users of varying levels of ability has not been envisioned until now, much less one focusing on running. Our approach could in future be extended to any sporting activity incorporating outdoor routes such as biking or hiking, while continuing to leverage a range of experience levels for all users. The domains which are suitable for this interactive approach are those where the object to be recommended itself has multiple facets. Also suitable are domains where the object is a composite of previous recommendations and we have chosen the route-planning task for those unfamiliar with their location, such as holiday-makers or those on business trips to an unfamiliar city.

The basic system architecture, presented in Figure 1, is composed of a number of key components as follows. First there is a case base of prior routes, each represented as a sequence of GPS location coordinates with the potential to include additional metadata (e.g. terrain information, difficulty level etc.) if available, scraped from a popular running website. The retrieval engine is responsible for selecting and ranking routes for recommendation, as described below. Finally, the adaptation and explanation interface provides users with an opportunity to adjust the recommended route to suit their exact needs as well as enhancing the recommended route by adding addition local information to help the user better understand the route being offered. We use libraries and technologies including Google Maps and photo service Panoramio in new ways, by using them to form the impetus by which a recommendation can be altered, using them to provide information on the surrounding area to make them aware of their options. The focus of the recommendation session for the user is exploring the space and altering the given waypoints so the system can adapt its current recommendation, making it an entirely interaction-focused experience.

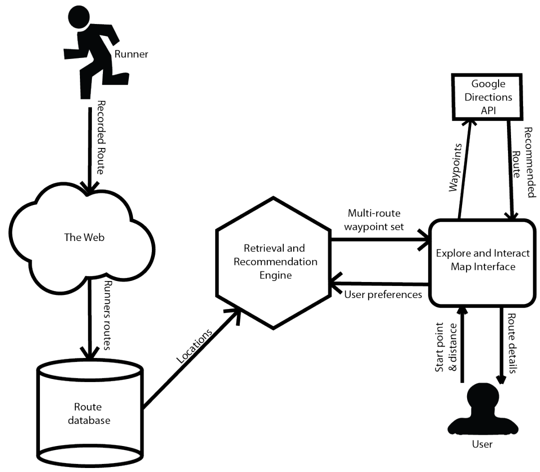

Figure 1: System architecture for Exercise Builder.

Retrieving Relevant Routes

The initial input from the user is a preferred starting point for the run and a preferred distance. Retrieving relevant routes from the case base involves a number of stages. First, available cases are filtered based on their proximity to the preferred starting point, so that we only consider routes that pass through, or within a 1 km. of the user's starting point. Next, we use the user's preferred distance as an upper-bound for case retrieval so that routes that are longer than the preferred distance are excluded from consideration. The reason for this upper-bound is that the routes are frequently circular in nature and cannot be effectively trimmed to a shorter length, as runners expect to return to where they started. This is a specific issue in the exercise domain that would make longer routes unsuitable, and as such is a non-trivial task future work can explore ways to address. This allows for a good recommendation to be selected, without the need for adaptation, as well as allowing for diverse routes to be selected based on their distance. In the case that no good matches are found potential cases are combined, producing routes of roughly the same distance that can be adapted based on user feedback.

In order to combine routes if necessary, *candidate* routes are sorted based on their *popularity* as follows. Each route is composed of a sequence of GPS coordinates, $Loc_1, \ldots, Loc_n$. In turn, a given location may be present in a number of different routes. Thus the popularity of Loc_i is the number of routes containing Loc_i (gien in equation 1) and the popularity of a given route is the sum of the popularity scores of its constituent locations, as in equation 2.

$$Popularity(Loc_i) = \{Route \in CB : Loc_i \in Route\} \quad (1)$$

$$Popularity(Route_i) = \sum Popularity(Loc_j) : \forall Loc_j \in Route_i \quad (2)$$

The routes that are selected for recommendation are those top-ranking routes whose aggregate distance is maximally close to the target route distance.

Composing Recommendations

At this point we have a set of candidate routes drawn from the area of interest and whose aggregate distance is within the distance requested by the user. Moreover, these routes traverse popular areas which improves their likely quality. A key challenge faced by the system now is how to assemble these multiple routes to produce a single coherent route. This is far from a simple matter of concatenating the selected routes. For a start there is no guarantee that these routes will overlap directly and arbitrarily concatenating them is fraught with issues. In the past much work has been done on route planning, especially on driving routes (ZITA et al. 1997). Many researchers have focused on algorithmic heuristics for combining these types of routes, often based on minimizing distance travelled (McGinty and Smyth 2001). These are problematic when applied to routes where the focus is not an optimization of travel from point A to point B. In running for example it is usual to finish where you started, and the criteria for a good route are more subjective, being based on environment and route difficulty due to terrain. This makes the use of a case base particularly important, in order to make use of the tacit knowledge veteran runners in the area have about good routes. Our approach, as distinct from the work in (McGinty and Smyth 2001) is that our approach is an attempt to optimize for popular run experiences in a situation where the participant expects to finish where they start and minimizing distance would be detrimental.

In this paper we adopt a different strategy by harnessing existing route-planning services such as the Google Maps route planner. To generate a consolidated route for final recommendation, the Google Maps system is used to generate a route based on a *waypoint specification* containing up to 8 waypoints, which produces a route that passes through each waypoint. To generate this specification we sample the 8 waypoints from the selected routes, choosing points that are evenly distributed within each candidate route pro rated for candidate route length. For example, if a user requests a 8km route in an area where no 8km routes exist, the system may retrieve a 5km, a 2km, and perhaps a 1km route. In this situation 5 waypoints will be drawn from the 5km case, 2 from the 2km case and a single waypoint from the final 1km case. These eight waypoints are chosen to be equally spaced along each route, and are combined using Google's DirectionService, which finds the path that touches them all. Our algorithm therefore works to create a composite route that is circular and suitable to travel in either direction, while being roughly their desired distance and giving the user sufficient information to decide if it is a good recommendation.

Explanation & Adaptation

In this research, recommendation – that is, the ability to recommend relevant routes – is just one part of a larger problem. In addition we should help users to better understand the suggestions that are being generated and, crucially, we should provide a facility to adapt the recommendation to better fit the needs that often go unexpressed via an original query, and that may be all but impossible for the user to express using traditional interaction. For this reason, as shown in Figure 2, the recommended route is displayed on a map-based interface.

In this scenario descriptive multimedia can play a part in informing users about possible choices. Rather than solely rely on route metadata or attributes, we make use of extrinsic data in our application. This means multimedia from the area surrounding and included in the routes without bias, which can prompt users to make decisions based on the area and not just on the available routes. This data is not the traditional explanation seen in CBR, rather it is an explanation of the surroundings, to give an indication to unfamiliar users of areas that might be of interest to them in the area, and what may make for a good experience on a run. This is reflected in not limiting the multimedia to the path of the route.

Of course given that the use-case for our research concerns a visitor to a new city, it is reasonable to assume that they will be interested in passing major sites and points of interest during their runs. For example, for a visitor to Beijing we may wish to highlight features along the recommended route such as the famous Bird's Nest stadium, or Tower Bridge for a London visitor, or the National Mall for a visitor to Washington DC. For this reason, the basic explanation component of this system is fulfilled by augmenting the route map with additional information to highlight points of interest. Once again we do this by relying on existing online resources, in this case by retrieving popular photos from the Panoramio photo-sharing site.

The map-based interface also provides for a simple and intuitive approach to extending route adaptation beyond the route composition algorithm by allowing users to tweak the recommended route via its waypoints. Recommended routes are therefore adapted by the system to make changes indicated by users through interaction. For example, the user can adjust the recommended route by dragging waypoints to new locations, perhaps to take in a point of interest that was by-passed.

The Exercise Builder System: A Demonstration

The above architecture was used to develop a prototype system which we call *Exercise Builder*. To create a case base of viable routes we imported more than 1,300 of the most popular routes from a popular run-tracking web service. The recommendation interface was developed using Google Maps and, as mentioned above, extrinsic explanation content was drawn from then Panoramio service, retrieving 50 images per route based on standard API access provisions.

In this demonstration of the Exercise Builder, we start with a default recommended route at the geographic centre of the imported routes for one city, Dublin, Ireland. The user

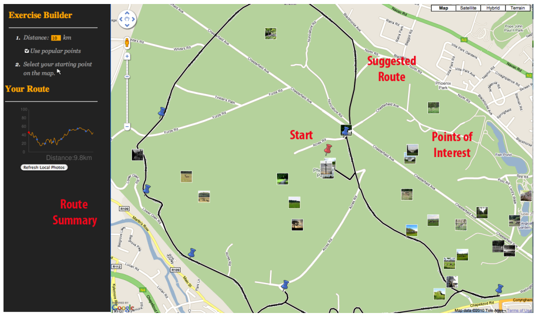

Figure 2: A recommended route including drag-points and multimedia content.

has the opportunity to provide their preferred running distance and move the starting point of the run to somewhere more suited to their needs, or simply start interacting with the provided waypoints.

These details are sent using AJAX to the server, which selects the points from a route that meet the user's requirements from the database. If this route is significantly too short (more than 1 km) the server accumulates additional points as previously mentioned and the result is 8 waypoints returned to the client which are then connected by Google's direction service to produce a coherent run from the most popular locations in suitable routes in the area (represented by the waypoints). This run forms the basis for adaptations made by the user, unless they modify the starting point of the run, at which point a new run is recommended based on their new starting point and any changes to target distance.

The initial recommendation based on the user's stated preferences looks similar to Figure 2. It is displayed like any other recommendation, along with route information like distance and elevation (taken from Google's Elevation API and rendered as a chart which shows the relative difficulty of the run). Thumbnails of Panoramio images are overlaid in the area around the route for the user to explore, the idea being to offer the user an option to move one of the route's waypoints and so generate a new route. Figure 3 shows a typical exploration, clicking on the thumbnail shows the larger version of the image in a pop-up. The routes begin and end at the user's desired start-point, which makes it easy for users to see the detours they can add by dragging waypoints to areas that draw their interest.

Summary and Future Work

In this paper we have presented a proof-of-concept application called Exercise Builder, a recommendation system designed to offer routing advice to joggers and runners in unfamiliar surroundings. The significance from a CBR perspective is that the system harnesses genuine user experiences (composite route plans) rather than simple atomic items. This in turn re-introduces the issue of adaptation as a key part of case-based recommendation and the system as presented provides for an interactive adaptation component via a map-based interface. We have also demonstrated the ability to complement case recommendations with supplementary information, in this case multimedia content drawn from additional online resources.

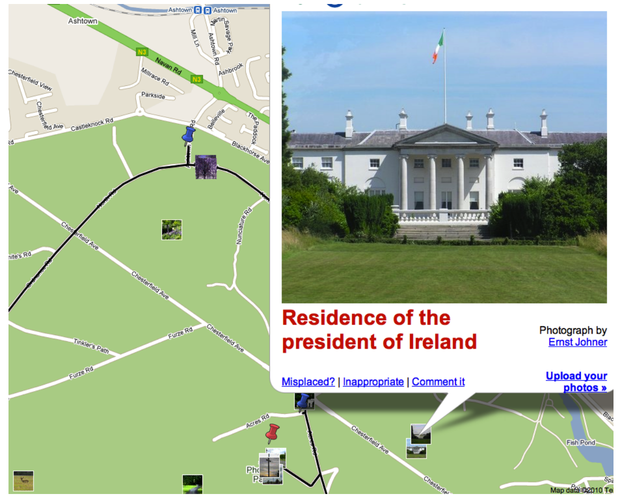

Figure 3: Multimedia content exploration in action.

There is clearly considerable opportunity for further work. We have had positive feedback from some 66 users who used the system, and intend to survey them fully. We are also experimentally evaluating the algorithm as it currently exists, as well as adding further features such as term weighting to the popularity metric used to further optimize resulting compound recommendations. There is a clear need to add additional routing or extrinsic meta-data to the case base and information about the terrain and difficulty of the routes is one obvious extension. In addition, the present system uses a simple query and simple case retrieval engine leaving many possibilities for alternatives. Certainly the availability of additional meta-data would provide an opportunity to improve the matching function. It may also be interesting to consider an alternative form of query specification, perhaps by allowing users to sketch a possible route of the map, which could then be refined by the recommender system. There is potential for the recommendation feedback and exploration to have ongoing impact on both immediate and future potential recommendations. It is also possible that a more immediate relevance-feedback style modification could occur whereby knowledge of a user can increase as the recommendation is built. This would combat the problems encountered with new users and build groupings of users quickly.

References

Alon, N.; Awerbuch, B.; Azar, Y.; and Patt-Shamir, B. 2009. Tell me who i am: An interactive recommendation system. *Theory of Computing Systems* 45:261–279. 10.1007/s00224-008-9100-7.

de Mántaras, R. L.; McSherry, D.; Bridge, D. G.; Leake, D. B.; Smyth, B.; Craw, S.; Faltings, B.; Maher, M. L.; Cox, M. T.; Forbus, K. D.; Keane, M. T.; Aamodt, A.; and Watson, I. D. 2005. Retrieval, reuse, revision and retention in case-based reasoning. *Knowledge Eng. Review* 20(3):215–240.

Knijnenburg, B. P.; Reijmer, N. J.; and Willemsen, M. C. 2011. Each to his own: how different users call for different interaction methods in recommender systems. In *Proceedings of the fifth ACM conference on Recommender systems*, RecSys '11, 141–148. New York, NY, USA: ACM.

McGinty, L., and Smyth, B. 2001. Collaborative case-based reasoning: Applications in personalised route planning. In *ICCBR*, 362–376.

McGinty, L., and Smyth, B. 2002. Shared experiences in personalized route planning. In *FLAIRS Conference*, 111–115.

Miyo, K.; Nittami, Y. S.; Kitagawa, Y.; and Ohe, K. 2007. Development of case-based medication alerting and recommender system: a new approach to prevention for medication error. *Stud Health Technol Inform* 129(Pt 2):871–874.

Pazzani, M. J., and Billsus, D. 2007. Content-based recommendation systems. In *The Adaptive Web*, 325–341.

Resnick, P., and Varian, H. 1997. Recommender systems. *Communications of the ACM* 40(3):58.

Shimazu, H. 2001. Expertclerk: navigating shoppers' buying process with the combination of asking and proposing. *Proceedings of the 17th international joint conference on Artificial intelligence - Volume 2* 1443–1448.

Smyth, B. 2009. The Sensor Web: Bringing Information to Life . *ERCIM News* 76:3.

Tunkelang, D. 2011. Recommendations as a conversation with the user. In *ACM RecSys 2011*.

ZITA, K.; JONATHAN, H.; SHEWCHUK, R.; and VELOSO, M. M. 1997. Exploiting domain geometry in analogical route planning. *Journal of Experimental & Theoretical Artificial Intelligence* 9(4):509–541.